

REMARKS

This Amendment is filed in response to the Office Action mailed on June 2, 2005.

All objections and rejections are respectfully traversed.

Claims 1-45 are in the case.

No Claims were amended.

No new Claims were added.

At Page 2 of the Office Action the Drawings were objected to as there being no figure designation on Sheet 4 of 6. Amendment of the drawing is believed to satisfy this objection. A substitute set of Formal Drawings is enclosed herewith.

At Paragraphs 1-2 of the Office Action Claims 41-45 were rejected under 35 U.S.C. 102(b) as being anticipated by Fuchs et al. U. S. Patent No. 5,440,726 issued August 8, 1995.(hereinafter Fuchs).

The present invention, as set forth in representative claim 41, comprises in part:

41. A file system, comprising:

- a backup memory storing a plurality of file system transaction entries;
- a first process that establishes *a swarm of messages with respect to the file system transaction entries and delivers the swarm of messages to the file system*;
- a second process that performs a LOAD phase in a concurrent manner for a plurality of messages in the swarm of messages; and
- a third process that performs a MODIFY phase for at least some messages in the swarm of messages, the MODIFY phase operating on messages based on the order in which file system transaction entries were stored in the backup memory.

Fuchs discloses item 82 in his Fig. 1 as::

“In a preferred embodiment, discussed below, the fault tolerant library 82 includes a critical memory function which allows a user to specify that certain data associated with an application process is critical data, which will be preferably stored by the fault tolerant computing system 5 in an area of critical memory 85.”

At Col 14 lines 11-32 Fuchs discloses:

“When a previously failed node returns to service, the watchdog 15 in that node obtains a copy of the node list 32 and fault tolerant process list 25 from the watchdog 15 in another node. The fault tolerant process list 25, discussed above in conjunction with Fig. 4a, indicates which nodes are currently executing the application process local to the previously failed node and the location of the state files containing the state of the local processes which are necessary to restart those processes. The watchdog 5 obtains copies of the files from the nodes currently executing the processes and restarts the processes using the state copies. As indicated above, when the watchdog 15 restarts a process, it sends a message to the other watchdogs 15 in the fault tolerant computing system 5, and if a watchdog 15 is running the restarted process, that watchdog 15 ceases running the process and modifies its fault tolerant process list 25 to indicate that the process is now running on the proper primary node. All of the other watchdogs 15 simply modify their fault tolerant process list 25 in the manner just indicated.”

Fuchs discloses at Col. 11, lines 3-19:

“For messages that were sent before the recovery line 310 and processed by the receiver after the recovery line 310, referred to as in-transit messages, such as messages M(4) and M(5), only the message content information in the logs are valid. The processing order for in-transit messages is either not logged or invalid. The in-transit messages are effectively still traveling in the communication channels and therefore can arrive in an arbitrary order due to unknown transmission delays. In-transit messages are the only messages that can be recorded.

Finally, messages sent after the recovery time 310, referred to as orphan messages, such as messages M(6) and M(7), are unsent and unreceived with respect to the recovery line. Thus, orphan messages are invalidated upon roll back and should be discarded by the receiving process otherwise the recovery is inconsistent.”

Further, Fuchs has been described in previous Amendments as follows:

In the Amendment filed on April 1, 2005,

“Fuchs teaches a computing system that concurrently executes a plurality of different application processes. *See* col. 5, line 66 through col. 6, line 4. The processes communicate with one another by passing messages. *See* col. 2, lines 38-42. Each application process is associated with a corresponding nonvolatile (backup) memory 44 containing logs of the process’s incoming and outgoing messages. *See* col. 6, lines 12-15 and fig. 1. The nonvolatile memory also stores the process’s “critical” program data, which is transferred to the memory at regular “checkpoint” time intervals. *See* col. 3, lines 2-5.”

In the Amendment filed on July 15, 2004,

“By way of background, Fuchs describes a progressive retry recovery system. Furthermore, Fuchs system is an in seriatim system because it replays messages in series, one after the other. (Col. 18, lines 36-44) The in seriatim technique was recognized by the Applicant's as a prior art technique and described by the applicant in the Background of the invention on page 4, lines 3-10 on the application as filed.”

In the Amendment filed on Jan. 26, 2004,

“Fuchs describes in the Abstract a progressive retry system based on checkpointing, message logging and message replaying. Fuchs describes the message replaying in further detail in column 18, lines 20-column 19, line 31. Fuchs does not teach or suggest establishment of a “*swarm* of messages with respect to the storage system transaction entries” (emphasis added) nor does it deliver the swarm to the file system, as claimed. Fuchs first restores the faulty process to the latest checkpoint (column 18, lines 25-28) and then replays the received messages since the last actual checkpoint in the receiver log file (column 18, lines 25-30). Fuchs does not teach the creation of a “*swarm* of messages,” but instead processes messages in *seriatum*. This was recognized by the Applicants as a prior art technique, and described by the Applicants in the Background of the Invention on page 4, lines 3-10 of the application as filed.”

Applicant respectfully urges that nowhere does Fuchs have any disclosure of Applicant's claimed novel *a swarm of messages with respect to the file system transaction entries and delivers the swarm of messages to the file system.*

Further, Applicant respectfully urges that Fuchs is completely silent concerning any processes to handle *a swarm of messages with respect to the file system transaction entries*, as claimed by Applicant.

The Examiner argues, at Paragraph 2 (Page 3) of the Office Action that Fuchs discloses, "a first process that establishes a swarm of messages with respect to the file system transaction entries" citing Fuchs' Abstract. Applicant respectfully urges that Fuchs' Abstract is silent concerning Applicant's claimed *swarm of messages with respect to the file system transaction entries*.

Further, the Examiner argues that Fuchs discloses "A second process that performs a load phase in a concurrent manner for a plurality of messages in the swarm of messages", citing Col. 14 lines 11-32. These lines were quoted hereinabove. Applicant respectfully urges that the cited, and Quoted, lines are totally silent concerning a swarm of messages as claimed by Applicant.

Still further, the Examiner argues that Fuchs discloses "A third process . . . for at least some messages in a swarm of messages" citing Col. 11 lines 3-19. Again, Applicant

respectfully urges that the cited, and quoted hereinabove, lines are silent concerning anything about a *swarm of messages with respect to the file system transaction entries*.

Accordingly, Applicant respectfully urges that Fuchs is legally precluded from anticipating Applicant's claimed novel invention because to the complete absence from Fuchs of Applicant's claimed novel *a swarm of messages with respect to the file system transaction entries and delivers the swarm of messages to the file system*.

At Page 3 (bottom) and Page 4 (top) of the Office Action Claims 1-4, 8-24 and 26-40 were rejected under 35 U.S.C. 103(a) as being unpatentable over Fuchs in view of Kelman U. S. Patent No. 6,671,820 December 30, 2003 (hereinafter Kelman)

The present invention, as set forth in representative claim 1, comprises in part:

"1. A system for replay of a backup memory in a storage system having a file system for managing transfer of data to and from an attached disk array, the system comprising:

 a log in the backup memory containing the storage system transaction entries accumulated after a consistency point at which time results of the storage system transaction entries are committed to the disk array;

an initiator process that establishes a swarm of messages with respect to the storage system transaction entries and delivers the swarm to the file system; and

a disk information-retrieval process in the file system that is carried out on the swarm of messages in parallel.

Kelman discloses using a logical unit masking driver to prevent a formerly crashed server from writing to dedicated backup storage devices on a computer network, as the formerly crashed server attempts to re-boot. Preventing writing to the dedicated backup devices prevents the server from possibly corrupting the backup storage devices.

Applicant respectfully urges that both cited patents, Fuchs and Kelman, are completely silent concerning Applicant's claimed novel *an initiator process that establishes a swarm of messages with respect to the storage system transaction entries . . . a disk information-retrieval process in the file system that is carried out on the swarm of messages in parallel.*

That is, Applicant respectfully urges that there is no disclosure in either Fuchs or Kelman concerning Applicant's claimed novel *a swarm of messages*.

Accordingly, Applicant respectfully urges that both Fuchs and Kelman, taken either singly or in combination, are legally precluded from rendering Applicant's claimed novel invention unpatentable under 35 U.S.C. 103 (a) because of the absence from both of Applicant's claimed novel *an initiator process that establishes a swarm of messages with respect to the storage system transaction entries . . .*

a disk information-retrieval process in the file system that is carried out on the swarm of messages in parallel.

At Paragraph 3 (Page 13) of the Office Action Claims 5 and 25 were rejected under 35 U.S.C. 103(a) as being unpatentable over Fuchs in view of Park U. S. Patent Application publication number 2003/0131190 published on July 10, 2003.

Applicant respectfully notes that Claims 5 and 25 are dependent claims, and are dependent from independent claims which are believed to be in condition for allowance.

At Paragraph 4 of the Office Action (Page 14) Claims 6 and 7 were rejected under 35 U.S.C. 103(c) as being unpatentable over Fuchs in view of Creighton U. S. Patent No. 6,330,570 issued on December 11, 2001 (hereinafter Creighton).

Applicant respectfully notes that Claims 6 and 7 are dependent claims, and are dependent from independent claims which are believed to be in condition for allowance.

All independent claims are believed to be in condition for allowance.

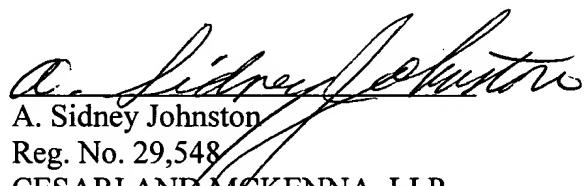
PATENTS
112056-0003
P01-1025

All dependent claims are dependent from independent claims which are believed to be in condition for allowance. Accordingly, all dependent claims are believed to be in condition for allowance.

Favorable action is respectfully solicited.

Please charge any additional fee occasioned by this paper to our Deposit Account No. 03-1237.

Respectfully submitted,


A. Sidney Johnston
Reg. No. 29,548
CESARI AND MCKENNA, LLP
88 Black Falcon Avenue
Boston, MA 02210-2414
(617) 951-2500